# 基于新的适应度函数和多搜索策略的高维多目标进化算法 *

代 才，石晓琪

(陕西师范大学 计算机科学学院, 西安 710119)

**摘 要：**如何平衡高维多目标进化算法的收敛性和多样性是一个困难且有挑战的工作。为提高高维多目标进化算法的性能，提出了一个基于新的适应度函数和多搜索策略的高维多目标进化算法。该算法提出了一个新的适应度函数来平衡多样性和收敛性，并且设计了一个多搜索策略来帮助交叉算子产生优秀的后代进而提高收敛性。该适应度函数首先从当前种群和新产生的后代中挑出收敛性较好的个体，然后计算这些个体的稀疏程度；该多搜索策略选择稀疏且收敛的解来执行全局和局部搜索。数值实验测试了 CEC2018 高维多目标竞赛的 15 个测试问题，每个测试问题的目标个数分别为 5、10、15。实验结果表明，该算法能找到一组比四种代表性算法（如 NSGAIII、MOEA/DD、KnEA、RVEA）具有更好的多样性和收敛性的解集。

**关键词：**高维多目标优化；适应度函数；多搜索策略；进化算法

**中图分类号：**TP391.9 doi: 10.19734/j.issn.1001-3695.2018.06.0485

# Many-objective evolutionary algorithm based on fitness function and multi-search strategy

Dai Cai, Si Xiaoqi

(*College of Computer Science Shaanxi Normal University*, *Xi'an* 710119, *China*)

**Abstract:** How to balance the convergence and diversity of many-objective evolutionary algorithms is a difficult and challenging work. This paper designs a many-objective evolutionary algorithm based on fitness function and multi-search strategy to improve the performance of many-objective evolutionary algorithms. The algorithm proposes a new fitness function to balance the convergence and diversity, and proposes a multi-search strategy to help crossover operators to generate good offspring. The fitness function firstly selects better convergent solutions from the current population and offspring, then calculates the sparsity of these convergent solutions. The multi-search strategy makes sparse and convergent solutions to be selected to carry out the exploration and exploitation. Experiments are conducted on fifteen benchmark functions with 5, 10, 15 objectives of CEC2018. The results indicate that the proposed algorithm can find a set of solutions with better diversity and convergence than four efficient state-of-the-art algorithms, e. g. , NSGAIII, MOEA/DD, KnEA, RVEA.

**Key words:** many-objective optimization; fitness function; multi-search strategy; evolutionary computation.



## 0 引言

在过去几十年中，提出了许多进化算法[1~3]来解决多目标优化问题。多目标进化算法使用种群进化策略来得到一组最优解决，是一种有效解决多目标优化问题的方法。在现实世界中，许多多目标优化问题[4~6]包含超过三个优化目标，这些问题一般称为高维多目标优化问题[7]。进化算法是一种基于种群进化的随机搜索算法，由于只需要计算目标函数值就能解决复杂的非线性问题而得到广泛的应用。但当处理高维多目标优化问题时，大部分现有的多目标进化算法的收敛性能会比较差。

在使用进化算法求解高维多目标优化问题时，现在需要面临两个问题：

a）随着目标维数的增加，构造近似帕累托前沿面所需解的个数呈指数级增长。如果种群规模比较大，所需的计算量就会比较大；如果种群规模比较小，非支配个体所占的比例随着目标维数的增加而迅速上升，这使得帕累托支配的选择压力越来越小，进而导致基于帕累托支配的多目标进化算法失去收敛性。

b）随着目标维数的增加，目标空间呈指数级增长，如果保持种群的多样性，个体之间的相关性就会降低，这会降低多目标进化算法的收敛速度；如果使种群中的个体之间的相关性比较强，这就无法保持种群的多样性。

收稿日期：2018-06-11；修回日期：2018-08-06 基金项目：国家自然科学基金资助项目（61502290，61401263，61672334/61673251）；中国博士后基金资助项目（2015M582606）

作者简介：代才（1984-），男，硕导，副教授，主要研究方向为进化计算、多目标优化（cdai0320@snnu.edu.cn）；石晓琪（1992-），女，硕士，主要研究方向为进化算法、决策优化.

## 1　相关研究

针对高维多目标优化问题的特点，国内外目前主要存在四大类方法来求解高维多目标优化问题。

a)降维法。这类方法的核心思想是通过降低目标函数的个数来增强算法的收敛压力。罗乃丽等人[8]通过衡量目标在近似解集上体现的冲突性,构造问题的冲突信息矩阵,对该矩阵进行特征分析,确定目标的重要性程度,进而实现目标函数的降维。Zou 等人[9]根据减少不同目标函数后的非支配解所占的比例的变化来确定要减少的目标函数。但这些降维法即使目标函数被缩减，剩余目标函数的个数依然可能超过三个，因此这类方法可以提高高维多目标演化算法的收敛速度，但不能从根本上解决高维多目标优化问题。

b)指标函数法。这类方法通过评价解集的质量（如多样性和收敛性）来指引算法的搜索。这类方法最具代表的算法是 Friedrich 等人[10]提出的基于指标函数的演化算法框架，使用指标函数来对每个个体进行排序，它要求这些指标具有“支配关系保持性”。由于大多问题的真实帕累托前沿面可能是未知的，目前能比较准确评价解集质量的度量方法只有超体积法和 R2[11]指标法。Falcon-Cardona 等人[11]将 R2 指标应用到蚁群算法中来解决高维多目标优化问题。Bader 等人[12]使用蒙特卡洛模拟法来近似快速计算超体积值，进而将它应用到基于指标函数的演化算法中来解决高维多目标优化问题。使用超体积作为指标函数的主要缺陷是计算超体积的复杂度会随着目标函数的个数呈指数级增加[13],即使使用近似方法估计超体积值，算法的时间复杂度依然非常高。R2 作为指标函数的缺陷是当面对帕累托前沿面比较复杂的问题时，找到一组合适的效用函数是比较困难的，这会直接影响基于 R2 的多目标演化算法的性能。虽然计算 R2 的时间复杂度低于计算超体积，但它的计算复杂度依然高于其他类型算法的计算复杂度，这些就限制了这类算法在高维多目标优化问题中的应用。

c)排序法。这类方法的核心思想是通过对个体进行排序来帮助算法增加选择压力。例如，Fleming 等人[14]将偏好信息应用到基于帕累托概念的排序法中，根据不断演化的结果逐步确定偏好，不断缩小决策者关注的区域，最后分离出决策者偏好的最优解；Wang 等人[15]提出了基于边界解（corner solutions）的排序方法；Dai 等人[16]通过改进 α 支配关系来平衡高维多目标演化算法的多样性和收敛性；林梦幔等人[17]将 α 支配关系应用到演化算法中来解决高维多目标优化问题。基于修改的帕累托支配法可以提高算法的收敛速度，但还需要另外的方法来维持解集的多样性。

d)基于分解的方法.这类方法使用分解的方法通过构造参考方向或者参考点，指引种群中的解分别向参考方向或者参考点靠近，进而增强算法的收敛性并能比较好地维持种解的多样性（参考点或参考方向的多样性好，大部分情况下种群多样性也会比较好）。例如，Deb 等人[18]通过设置一组参考点作为偏好的最优解集，然后指引非支配解分别向这些参考解靠近；Yuan 等人[19]根据一组参考点对当前种群进行分类，对同一类中的解，使用聚合函数来对解进行排序，使种群中的解分别向参考点靠近，进而实现提高收敛性和保持多样性的目的。另外一些改进的基于分解的多目标演化算法[20]也被提出来解决高维多目标优化问题[21,22]。

笔者认为通过产生优秀的后代，解集的多样性和收敛性都可以被提高。为实现这个目标，首先设计了一个多搜索策略帮助交叉算子执行全局和局部搜索，该搜索策略选择稀疏且收敛性好的个体作为父代个体来增强搜索效率。另外设计了一个新的适应度函数来平衡种群的收敛性和多样性。该适应度函数首先从当前种群和新产生的后代中挑出收敛性较好的个体，这有助于提高种群的收敛性；然后根据目标向量间的二范数和无穷范数的乘积来删除拥挤的个体，这有助于维持种群的多样性。

## 2　高维多目标优化的相关概念

在本章中首先介绍高维多目标优化的主要概念。一个高维多目标优化问题[7]可以由式（1）表示：

$$\begin{cases} \min F(x) = (f_1(x), f_2(x), \cdots, f_m(x)) \\ \text{s.t.} \quad g_i(x) \le 0, i = 1, 2, \cdots, q \\ \quad h_j(x) = 0, j = 1, 2, \cdots p \end{cases} \tag{1}$$

其中：$x = (x_1, \cdots, x_n) \in R^n$ 是一个 $n$ 维变量；$f_i(x)(i = 1, \cdots, m)$ 是目标函数，$m$ 是目标函数的个数且大于 3；$g_i(x)(i = 1, 2 \cdots q)$ 是 $q$ 个不等式约束；$h_j(x)(j = 1, 2, \cdots p)$ 是 $p$ 个等式约束，并且满足所有约束的可行解组成的集合定义为 $\Omega$ 。

下面介绍四个高维多目标优化的重要概念：a）对于给定的两个解 $x, z \in \Omega$ ，如果 $f_i(x) \le f_i(z)(i = 1, \cdots, m)$ 且 $\|F(x) - F(z)\|_2 \ne 0$ ，那么就称 $x$ 支配 $z$ （或者定义为 $x \prec z$ ）；b）如果 $x$ 不被其他任何解支配，那么 $x$ 就是帕累托最优解（Pareto optimal solution）；c）所有帕累托最优解组成的集合称为帕累托最优解集（set of Pareto optimal solutions (PS)）；d） 所有帕累托最优解的目标向量组成的集合称为帕累托最优前沿面（Pareto optimal front (PF)）。

## 3　基于新的适应度函数和多搜索策略的高维多目标进化算法

为了能更好地平衡高维多目标进化算法的收敛性和多样性，一个新的适应度函数被设计来维持种群的多样性和增加种群的收敛压力，并且使用一个多搜索策略被来帮助交叉算子产生优秀的后代，进而有助于提高种群的多样性和收敛性。在本章中将会逐一地介绍新的适应度函数和多搜索策略。

### 3.1　新的适应度函数

新的适应度函数的主要目标是帮助算法维持种群的多样性和增加种群的收敛压力。为了计算这个适应度函数，需要一组均匀分布的权重向量 $W = \{\lambda^1, \lambda^2, \cdots, \lambda^N\}$ 。这里 $N$ 是权重向量的个数；$\lambda^i = (\lambda_1^i, \cdots, \lambda_m^i)(i = 1, \cdots, N)$ 。对于要计算适应度函数值的

种群 $POP=\{x^1,x^2,\cdots x^{NM}\}$，首先用式（2）找出每个权重向量对应的最好个体。

$$P=\{\arg\min_{x\in POP}\Delta\left(F(x),\lambda^i\right),i=1,\cdots,N\} \tag{2}$$

其中：$\Delta\left(F(x),\lambda^i\right)=\max_{1\le i\le m}\{|f_i(x)-Z_i|/\lambda_i^i\}$ 是一个切比雪夫函数，这里 $Z=(Z_1,\cdots Z_m)$ 是一个参考点，且 $Z_i=min\{f_i(x)|x\in\Omega\}$。$\min_{x\in\Omega}\Delta\left(F(x),\lambda^i\right)$ 的最优解是式（1）的一个帕累托最优解。其次用 $DP=\{F(x)-Z_2\,|\,x\in P\}$ 个体收敛性的阈值，并且用式（3）计算每个个体的拥挤度。

$$d(x)=\min\{F(x)-F(y)_2*F(x)-F(y)_\infty\,|\,y\in POP\cap y\ne x\} \tag{3}$$

$F(x)-F(y)_2$ 表示两个解的欧式距离；$F(x)-F(y)_\infty$ 表示两个解的最大差异性。最后,通过式(4)计算每个个体的适应度值。

$$\text{Fit}(x)=\begin{cases}2*\max_{y\in POP}d(y)+d(x) & \text{if}\quad F(x)-Z_2\le DP\\ d(x) & \text{else}\end{cases} \tag{4}$$

个体的适应度值越大，个体就越好。条件 $F(x)-Z_2\le DP$ 可以增强算法的选择压力，进而提高收敛性。

这个适应度函数的基本原理是：使用给定的权重向量和切比雪夫函数将高维多目标优化问题转换成一组单目标优化问题，在进化的每一代，可以找出每个单目标优化问题的当前最优解，然后用 $DP$ 来判断其余解的收敛性，最后用式（3）度量每个解的稀疏程度。这个适应度函数既可以度量解的收敛性，也可以度量解的稀疏程度。

### 3.2 多搜索策略和交叉算子

搜索策略在高维多目标进化算法中起到一个非常重要的作用。一个合适的搜索策略可以帮助交叉算子执行全局搜索和局部搜索。在本文中使用一个多搜索策略来平衡算法的全局和局部搜索。在介绍这个多搜索策略之前，首先计算任意两个权重向量间的欧氏距离，并且找出每个权重向量的 $T$ 个最近权重向量。用 $B(i)=\{i_1,\cdots,i_T\}$ 表示记录权重向量 $\lambda^i$ 的 $T$ 个最近邻权重向量，其中 $\lambda^{i_1},\cdots\lambda^{i_T}$ 就是 $\lambda^i$ 的 $T$ 个最近邻权重向量。第一个搜索策略是帮助交叉算子执行局部搜索。这里所使用的交叉算子（差分进化[23]）如式（5）所示。

$$x_j^{new}=\begin{cases}x_j+L\left(x_j^{r1}-x_j^{r2}\right) & \text{if rand}(0,1)<CR\\ x_j & \text{otherwise}\end{cases} \tag{5}$$

这里 $x=\{x,\cdots,x_n\}$ 是式（2）P 中的一个解；$j=1,\cdots,n$，$x_j^{r1}$ 是 $x^{r1}$ 的第 $j$ -th 个分量；假设 $\lambda^t$ 是 x 对应的权重向量，$r1$ 和 $r2$ 是 $B(t)$ 中的两个随机数，且 $x^{r1}=\arg\min_{x\in POP}\Delta\left(F(x),\lambda^{r1}\right)$，$x^{r2}=\arg\min_{x\in POP}\Delta\left(F(x),\lambda^{r2}\right)$；$L\in[0,2]$ 是一个控制搜索步长的比例因子；$CR$ 是交叉概率；$x^{new}=(x_1^{new},x_2^{new},\cdots,x_n^{new})$。这个搜索策略是从收敛性比较好的集合 P 中选择父代个体，并且这些父代个体互为邻居，这就有助于式（5）执行局部搜索并能加快收敛速度。

第二个搜索策略是帮助交叉算子式（5）执行全局搜索。首先用式（4）计算当前种群中每个个体的适应度值，然后用赌轮选择法选择若干个较好的个体，式（5）中的 $x=\{x,\cdots,x_n\}$ 就是这些较好个体中的一个，$x^{r1}$ 和 $x^{r2}$ 是当前种群中任意两个不同解。由于 $x^{r1}$ 和 $x^{r2}$ 是任意选的，$x^{r1}-x^{r2}$ 的值就会比较大，这就有利于进行全局搜索。

### 3.3 算法步骤

基于前面的内容，提出了一个基于新的适应度函数和多搜索策略的高维多目标进化算法（many-objective evolutionary algorithm based on fitness function and multi-search strategy, MaOEA-FM）。这个算法的伪代码如下所示：

算法　MaOEA-FM 的伪代码

输入：

MaOP (1)；停止条件；

$N$：权重向量的个数和种群 $POP$ 的规模；

$T$：每个权重向量的邻居数，$0<T<N$；

$\lambda^1,\lambda^2,\cdots,\lambda^N$：$N$ 个均匀分布的权重向量。

输出：近似的帕累托最优前沿面：$F(POP)$。

初始化：随机地产生一个初始种群 $POP=\{x^1,x^2,\cdots x^N\}$；确定参考点 $Z=(z_1,\cdots,z_m)$；确定 $B(i)=\{i_1,\cdots,i_T\},(i=1,\cdots,N)$，这里 $\lambda^{i_1},\cdots,\lambda^{i_T}$ 是 $\lambda^i$ 的 $T$ 个最近邻权重向量。

while 停止条件不满足时执行

令 $O=\varnothing$；算出每个权重向量对应子问题的当前最优解，用交叉算子式（5）和第一种搜索策略产生 $N$ 个新解，并将这些新解放入 $O$ 中；用式（4）计算 $POP$ 中每个个体的适应度值，用赌轮选择法选择 $N$ 个较好个体，对每个较好个体，使用第二种搜索策略和交叉算子式（5）产生新解，并将这些新解放入 $O$ 中。

更新 $Z$：For $j=1,\cdots,m$，if $z_j\langle\min\{f_j(x^{new})|x^{new}\in O\}$，令 $z_j=f_j\left(x^{new}\right)$。

计算 $O\cup POP$ 中每个的适应度函数值，令 $POP=\varnothing$，挑出 $N$ 个适应度值最大的个体，并将这些个体放入 $POP$ 中。

end while

在这个工作中，每个搜索策略产生一半的后代。

## 4 数值实验

### 4.1 测试问题和参数设置

本文使用 CEC2018[24]高维多目标竞赛上的测试问题来验证所提算法的有效性。　CEC2018 高维多目标竞赛一共包含 15 个测试问题，这些测试问题具有多样的特性，可以很好地测试高维多目标进化算法的优点和缺陷。四种有效的高维多目标进化算法（如 NSGAIII[18]、MOEA/DD[25]、KnEA[26]、RVEA[27]）被用来与所提算法进行对比。这四种算法的源代码都是来自 PlatEMO[28],它们的参数设置也采用 PlatEMO 上的默认设置。在






MaOEA-FM 中，$CR = 0.5$，$L = 0.5$，$T = 10$。其他的实验参数设置（如种群规模、最大函数评估次数、算法独立运行次数、测试问题的维数和变量的范围等）与 CEC2018 高维多目标竞赛上的设置相同，详细情况可以参见文献[24]。

### 4.2 性能指标

在本实验中笔者使用逆世代距离[29]（inverted generational distance，IGD）来定量度量算法的性能。IGD 可以综合度量一个种群的收敛性和多样性。IGD 的值越小，被度量种群的多样性和收敛性就越好。每个测试问题都在帕累托最优前沿面上均匀地采样 10 000 个点来计算 IGD 的值。另外，用 Wilcoxon Rank-Sum test [30]来度量算法的显著性统计。这里显著性水平设置为 0.05。(“+”), (“=”), 和 (“-”)分别表示 MaOEA-FM 的性能优于、等于和劣于对比的算法。

### 4.3 数值实验结果

附表 1 显示了 MaOEA-FM 和四种对比算法在 CEC2018 高维多目标竞赛问题上得到的 IGD 度量值。其中 F1-k 表示高维多目标问题，F1 的目标个数是 k。最好性能的结果用黑色标记出来了。

从附表 1 可以看到，对于 45 个测试问题，MaOEA-FM 分别在38、40、36、39 个问题上取得IGD度量值显著优于NSGAIII、MOEA/DD、KnEA、RVEA。这表明在大部分问题上，MaOEA-FM 所得到种群的质量要优于 NSGAIII、MOEA/DD、KnEA、RVEA 得到种群的质量。对于帕累托最优前沿面不能完全覆盖单位超平面的八个问题（F1、F2、F4、F7、F8、F9 和 F15），MaOEA-FM 所得到的 IGD 均值分别在 22、21、21、21 个问题上小于 NSGAIII、MOEA/DD、KnEA、RVEA 所得到的 IGD 均值；对于帕累托最优前沿面完全覆盖单位超平面的六个问题（F3、F10-F14），MaOEA-FM 所得到的 IGD 均值分别在 15、16、14、15 个问题上小于 NSGAIII、MOEA/DD、KnEA、RVEA 所得到的 IGD 均值。这些对比结果说明，该算法的适应度函数对于具有不同类型帕累托最优前沿面的问题也可以很好地维持种群的多样性和收敛性。对于具有许多局部帕累托最优前沿面的三个问题（F3、F4 和 F7），MaOEA-FM 在大部分问题上比 NSGAIII、MOEA/DD、KnEA、RVEA 具有更好的收敛性，这间接说明算法所使用的多搜索策略可以帮助算法提高搜索效率。

## 5 结束语

针对现有多目标进化算法处理高维多目标优化问题时所遇到的问题，本文提出了一种基于新的适应度函数和多搜索策略和的进化算法。在该算法中，设计了一个新的适应度函数来维持种群的多样性和帮助算法提高收敛性，采用了一个多搜索策略来平衡全局搜索和局部搜索。并且与最新的且比较有效的高维多目标进化算法进行对比，在 CEC2018 高维多目标竞赛问题上验证了所提算法的有效性。在将来的工作中，将继续研究该算法求解现实中高维多目标优化问题的性能。

## 参考文献：


[1] Cheng Ran, Rodemann F T M, Olhofer M, *et al.* Evolutionary many-objective optimization of hybrid electric vehicle control: from general optimization to preference articulation [J]. IEEE Trans on Emerging Topics in Computational Intelligence, 2017, 1 (2): 97-111.

[2] Zuo Liyun, Shu Lei, Dong Shoubin, *et al.* A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing [J]. IEEE Access, 2014, 3: 2687-2699.

[3] Zhang Yong, Gong Dunwei, Cheng Jian. Multi-objective particle swarm optimization approach for cost-based feature selection in classification [J]. IEEE//ACM Trans on Computational Biology and Bioinformatics, 2017, 14 (1): 64-75.

[4] Narukawa K, Rodemann T. Examining the performance of evolutionary many-objective optimization algorithms on a real-world application [C]// Proc of the 6th International Conference on Genetic and Evolutionary Computing. 2012: 316-319.

[5] Lygoe R J, Cary M, Fleming P J. A real-world application of a many-objective optimisation complexity reduction process [C]// Proc of the 7th International Conference on Evolutionary MultiCriterion Optimization. 2013: 641-655.

[6] Liu Hai-Lin, Chen Lei, Zhang Qingfu, *et al.* Adaptively allocating search effort in challenging many-objective optimization problems [J]. IEEE Trans. Evolutionary Computation, 2018, 22 (3): 433-448.

[7] Hughes E J. Evolutionary many-objective optimisation: many once or one many? [C]// Proc of IEEE Congress on Evolutionary Computation. 2005: 222–227.

[8] 罗乃丽，李霞，王娜. 利用冲突信息降维的进化高维目标优化算法 [J]. 信号处理, 2017, 33 (9): 1169-1178. (Luo Nai-li, Li Xia, Wang Na. Objective reduction using conflict information in many-objective optimization evolutionary algorithm [J]. Journal of Signal Processing, 2017, 33 (9): 1169-1178. )

[9] Zou Juan, Zheng Jinhua, Shen Ruiming, *et al.* A novel metric based on changes in Pareto domination ratio for objective reduction of many-objective optimization problems [J]. Journal of Experimental & Theoretical Artificial Intelligence, 2017, 29 (5): 983-994.

[10] Friedrich T, Horoba C, Neumann F. Multiplicative approximations and the hypervolume indicator [C]// Proc of Genetic and Evolutionary Computation Conference. 2009: 571-578.

[11] Falcon-Cardona J G, Coello C A. A new indicator-based many-objective ant colony optimizer for continuous search spaces [J]. Swarm Intelligence, 2017, 11 (1): 71-100.

[12] Bader J, Zitzler E. HypE: an algorithm for fast hypervolume-based many-objective optimization [J]. Evolutionary Computation, 2011, 19: 45-76.

[13] Beume N, Fonseca C, López-Ibáñe M, *et al.* On the complexity of computing the hypervolume indicator [J]. IEEE Trans on Evolutionary Computation,








2009, 13 (5): 1075-1082.

[14] Fleming P J, Purshouse R C, Lygoe R J. Many-objective optimization: an engineering design perspective. evolutionary multi-criterion optimization [M]. Berlin: Springer-Verlag, 2005: 14-32.

[15] Wang Handing, Yao Xin. Corner sort for Pareto-based many-objective optimization [J]. IEEE Trans on Cybernetics, 2014, 44 (1): 92-102.

[16] Dai Cai, Wang Yuping, Hu Lijuan. An improved alpha-dominance strategy for many-objective optimization problems [J]. Soft Computing, 2016, 20 (3): 1105-1111.

[17] 林梦嫚，周欢，王丽萍．基于 alpha 支配的高维目标进化算法研究 [J]. 计算机科学，2017, 44 (1): 264-270. (Lin Mengman, Zhou Huan, Wang Liping. Research of many-objective evolutionary algorithm based on Alpha dominance [J]. Computer Science, 2017, 44 (1): 264-270. )

[18] Deb K, Jain H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: solving problems with box constraints [J]. IEEE Trans on Evolutionary Computation, 2014, 18 (4): 577-601.

[19] Yuan Yuan, Xu Hua, Wang Bo, *et al.* A new dominance relation-based evolutionary algorithm for many-objective optimization [J]. IEEE Trans on Evolutionary Computation, 2016, 20 (1): 16-37.

[20] Zhang Qingfu, Li Hui. MOEA//D: a multiobjective evolutionary algorithm based on decomposition [J]. IEEE Trans on Evolutionary Computation, 2007, 11 (6): 712-731.

[21] Zheng Wei, Tan Yanyan, Fangxiaonan, *et al.* An improved MOEA//D with optimal DE schemes for many-objective optimization problems [J]. Algorithms, 2017, 10 (3): 86.

[22] Ishibuchi H, Doi K, Nojima Y. On the effect of normalization in MOEA//D for multi-objective and many-objective optimization [J]. Complex & Intelligent Systems, 2017, 3 (4): 279-294.

[23] Price K, Storn R M, Lampinen J A. Differential evolution: a practical approach to global optimization, ser. natural computing [M]. Secaucus, NJ, USA: Springer-Verlag, 2005.

[24] Cheng Ren. Benchmark functions for CEC'2018 competition on manyobjective optimization [R]. Birmingham: School of Computer Science, University of Birmingham Edgbaston, 2017.

[25] Li Ke, Deb K, Zhang Qingfu, *et al.* An evolutionary many-objective optimization algorithm based on dominance and decomposition [J]. IEEE Trans on Evolutionary Computation, 2015, 19 (5): 694-716.

[26] Zhang Xingyi, Tian Ye, Jin Yaochu. A knee point driven evolutionary algorithm for many-objective optimization [J]. IEEE Trans on Evolutionary Computation, 2015, 19 (6): 761-776.

[27] Cheng Ren, Jin Jaochu, Olhofer M, *et al.* A reference vector guided evolutionary algorithm for many-objective optimization [J]. IEEE Trans on Evolutionary Computation, 2016, 20 (5): 773-791.

[28] Tian Ye, Cheng Ren, Zhangxingyi, etal. PlateEMO: a MATLAB platform for evolutionary multi-objective optimization [J]. IEEE Comput. Intell. Mag. , 2017, 12 (4): 73-87.

[29] Zitzler E, Thiele L, Laumanns M, *et al.* Performance assessment of multiobjective optimizers: an analysis and review [J]. IEEE Trans on Evolutionary Computation, 2003, 7 (2): 117-132.

[30] Robert S, Torrie J, Dickey D. Principles and procedures of statistics: a biometrical approach [M]. New York: McGraw-Hill, 1997.

## 附录

表 1　MaOEA-FM、NSGAIII、MOEA/DD、KnEA 和 RVEA 得到 IGD 度量值的均值和方差

Table 1　Mean and standard deviation values of IGD obtained by MaOEA-FM, NSGAIII, MOEA/DD, KnEA and RVEA

| | MaOEA-FM | NSGAIII | MOEA/DD | KnEA | RVEA |
|---|---|---|---|---|---|
| F1-5 | **1.1077e-1(1.71e-4)** | 2.0756e-1 (5.91e-3)(+) | 2.9017e-1 (2.80e-2)(+) | 1.2371e-1 (1.83e-3)(+) | 4.0018e-1 (1.75e-1)(+) |
| F2-5 | **9.2262e-2(1.95e-4)** | 1.2910e-1 (4.25e-4)(+) | 1.3664e-1 (3.38e-3)(+) | 1.2766e-1 (3.92e-3)(+) | 1.2787e-1 (1.53e-3)(+) |
| F3-5 | **7.6573e-2(1.51e-3)** | 9.6731e-2 (1.80e-3)(+) | 1.1809e-1 (1.43e-3)(+) | 1.5308e-1 (6.73e-2)(+) | 8.2950e-2 (7.68e-3)(+) |
| F4-5 | **2.8536+0(9.18e-2)** | 3.0035e+0 (2.56e-1)(+) | 7.6920e+0 (3.22e-1)(+) | 2.9191e+0 (3.14e-1)(+) | 4.9814e+0 (1.38e+0)(+) |
| F5-5 | **2.1727e+0(9.56e-3)** | 2.2085e+0 (5.09e-4)(+) | 6.3810e+0 (3.09e-1)(+) | 2.4229e+0 (4.43e-2)(+) | 2.3478e+0 (1.73e-1)(+) |
| F6-5 | **3.0330e-3(1.38e-6)** | 9.1571e-2 (2.57e-2)(+) | 7.9096e-2 (4.66e-4)(+) | 6.9374e-3 (1.65e-3)(+) | 9.3646e-2 (2.29e-2)(+) |
| F7-5 | **2.4269e-1(2.62e-3)** | 3.2498e-1 (1.27e-2)(+) | 3.0005e+0 (1.37e-6)(+) | 2.9897e-1 (3.51e-3)(+) | 4.4937e-1 (3.54e-3)(+) |
| F8-5 | **2.0046e-1(4.00e-4)** | 2.1374e-1 (4.99e-3)(+) | 3.0704e-1 (1.95e-2)(+) | 2.6409e-1 (2.90e-2)(+) | 4.4649e-1 (7.36e-2)(+) |
| F9-5 | **1.2424e-1(1.55e-4)** | 2.6409e-1 (5.47e-3)(+) | 2.4667e-1 (3.57e-3)(+) | 7.8574e-1 (1.29e-1)(+) | 3.8989e-1 (8.60e-2)(+) |
| F10-5 | 5.2319e-1(2.78e-2) | 4.6073e-1 (1.46e-2)(-) | 7.7469e-1 (1.54e-1)(+) | 5.2086e-1 (3.94e-2)(=) | **4.4844e-1 (1.51e-2)(-)** |
| F11-5 | 1.6214e+0(8.29e-2) | 8.2588e-1 (3.81e-3)(-) | 4.5150e+0 (6.44e-2)(+) | **6.5471e-1 (1.16e-1)(-)** | 1.7788e+0 (5.67e-1)(+) |
| F12-5 | **1.0295e+0(3.72e-3)** | 1.1169e+0 (5.17e-3)(+) | 1.2902e+0 (2.17e-2)(+) | 1.1662e+0 (9.79e-3)(+) | 1.1274e+0 (1.00e-2)(+) |
| F13-5 | **1.0222e-1(1.70e-2)** | 2.3375e-1 (7.27e-3)(+) | 2.6117e-1 (3.68e-2)(+) | 2.2829e-1 (7.28e-3)(+) | 6.6361e-1 (1.41e-1)(+) |
| F14-5 | **3.7145e-1(7.58e-2)** | 7.0040e-1 (2.44e-1)(+) | 3.8364e-1 (5.18e-2)(+) | 4.8626e-1 (5.30e-2)(+) | 8.3614e-1 (1.70e-1)(+) |
| F15-5 | 7.0647e-1(3.05e-2) | 1.1500e+0 (9.48e-3)(+) | **5.7556e-1 (2.44e-2)(-)** | 3.7846e+0 (2.20e+0)(+) | 6.0225e-1 (4.49e-2)(-) |
| F1-10 | **2.2146e-1(3.02e-2)** | 2.7014e-1 (5.94e-3)(=) | 3.9192e-1 (2.21e-2)(+) | 2.3097e-1 (5.20e-3)(+) | 6.5904e-1 (7.38e-2)(+) |
| F2-10 | **1.5247e-1(9.09e-3)** | 2.0240e-1 (1.56e-2)(+) | 2.2270e-1 (4.80e-3)(+) | 1.6817e-1 (1.37e-2)(+) | 2.6052e-1 (7.97e-2)(+) |
| F3-10 | **1.1426e-1(3.98e-1)** | 2.7372e+2 (1.68e+2)(+) | 1.1529e-1 (2.30e-3)(+) | 2.3297e+6 (1.86e+6)(+) | 1.9320e-1 (2.20e-1)(+) |
| F4-10 | **7.7648e+1(7.16e-1)** | 8.8824e+1 (2.60e+0)(=) | 4.0451e+2 (7.49e+0)(+) | 8.0210e+1 (8.89e+0)(+) | 2.0455e+2 (4.12e+1)(+) |
| F5-10 | **5.6978e+1(2.79e+0)** | 8.9061e+1 (2.65e-1)(+) | 2.9689e+2 (1.23e+0)(+) | 7.3017e+1 (5.21e-1)(+) | 9.9679e+1 (6.17e+0)(+) |

| | MaOEA-FM | NSGAIII | MOEA/DD | KnEA | RVEA |
|---|---|---|---|---|---|
| F6-10 | **3.0489e-3(2.16e-6)** | 2.9435e-1 (7.45e-2)(+) | 1.1321e-1 (3.54e-3)(+) | 1.3068e+1 (4.24e+0)(+) | 1.2161e-1 (1.75e-2)(+) |
| F7-10 | 1.2147e+0(1.55e-2) | 1.0316e+0 (9.09e-2)(-) | 2.9641e+0 (6.69e-2)(+) | **8.6974e-1 (9.59e-3)(-)** | 2.9783e+0 (3.07e-1)(+) |
| F8-10 | **1.4069e-1(4.05e-4)** | 3.3710e-1 (1.24e-2)(+) | 9.0462e-1 (7.04e-3)(+) | 1.6873e-1 (3.32e-2)(+) | 7.7009e-1 (1.13e-1)(+) |
| F9-10 | **1.9014e-1(1.76e-3)** | 6.2161e-1 (1.18e-1)(+) | 4.4369e-1 (4.18e-3)(+) | 4.7242e+1 (4.44e+1)(+) | 7.8129e-1 (2.29e-1)(+) |
| F10-10 | 1.2614e+0(3.81e-2) | **1.0680e+0 (2.53e-2)(-)** | 1.9032e+0 (1.25e-1)(+) | 1.1648e+0 (7.41e-2)(-) | 1.2659e+0 (6.46e-2)(+) |
| F11-10 | **1.4387e+0(9.18e-1)** | 4.3444e+0 (9.03e-2)(+) | 1.5214e+0 (2.89e-2)(+) | 2.7405e+0 (5.88e-1)(+) | 8.8117e+0 (1.82e+0)(+) |
| F12-10 | **4.1596e+0(4.92e-3)** | 4.5939e+0 (8.38e-3)(+) | 6.6451e+0 (1.17e-1)(+) | 4.5421e+0 (1.12e-2)(+) | 4.5082e+0 (6.74e-2)(+) |
| F13-10 | **1.4915e-1(2.79e-2)** | 2.1097e-1 (1.32e-2)(+) | 3.0229e-1 (1.74e-2)(+) | 1.5108e-1 (1.41e-2)(=) | 9.5107e-1 (2.16e-1)(+) |
| F14-10 | 6.6348e-1(1.39e-1) | 1.6855e+0 (3.11e-1)(+) | **5.2303e-1 (2.26e-2)(-)** | 4.5108e+1 (2.78e+1)(+) | 6.0231e-1 (2.88e-2)(-) |
| F15-10 | 1.3627e+0(1.57e-1) | 1.6332e+0 (2.60e-1)(-) | 9.8570e-1 (8.70e-3)(-) | 6.6282e+0 (3.63e+0)(+) | **9.5090e-1 (6.06e-2)(-)** |
| F1-15 | **2.3617e-1(3.53e-2)** | 3.1410e-1 (1.20e-4)(-) | 5.2549e-1 (5.74e-3)(+) | 2.7435e-1 (1.72e-3)(+) | 7.0604e-1 (7.76e-2)(+) |
| F2-15 | **1.8143e-1(1.85e-3)** | 2.0260e-1 (4.33e-4)(+) | 3.9623e-1 (1.46e-3)(+) | 1.8233e-1 (1.41e-3)(=) | 5.3873e-1 (2.32e-1)(+) |
| F3-15 | 1.2385e-1(4.17e-4) | 1.8318e-1 (4.09e-2)(+) | 1.1059e-1 (1.87e-3)(-) | 4.9121e+9 (5.03e+9)(+) | **9.5038e-2 (5.67e-3)(-)** |
| F4-15 | **4.0428e+3(2.21e+1)** | 4.1050e+3 (1.99e+2)(+) | 1.5425e+4 (1.10e+3)(+) | 1.4869e+4 (2.14e+4)(+) | 8.4500e+3 (1.96e+3)(+) |
| F5-15 | **1.4168e+3(2.03e+1)** | 2.6072e+3 (1.33e+1)(+) | 7.3113e+3 (8.35e+0)(+) | 1.5267e+3 (6.25e+1)(+) | 3.3958e+3 (7.04e+2)(+) |
| F6-15 | **2.9561e-3(5.13e-7)** | 3.2105e-1 (3.01e-2)(+) | 1.3327e-1 (1.88e-3)(+) | 2.0123e+1 (3.29e+0)(+) | 2.4134e-1 (1.81e-1)(+) |
| F7-15 | **2.1564e+0(6.24e-1)** | 4.3214e+0 (4.75e-1)(+) | 3.4471e+0 (2.51e-2)(+) | 2.4327e+0 (1.54e-1)(+) | 4.1309e+0 (6.74e-1)(+) |
| F8-15 | 1.7326e-1(1.85e-2) | 4.1812e-1 (4.99e-3)(+) | 1.3421e+0 (3.91e-3)(+) | **1.3876e-1 (1.13e-3)(-)** | 1.2975e+0 (3.08e-1)(+) |
| F9-15 | **1.9541e-1(5.45e-3)** | 1.4357e+0 (1.51e+0)(+) | 9.6572e-1 (2.15e-3)(+) | 9.5128e-1 (6.43e-1)(+) | 1.2312e+0 (2.31e-1)(+) |
| F10-15 | **1.4687e+0(9.89e-2)** | 1.5352e+0 (6.98e-3)(+) | 2.3803e+0 (2.88e-2)(+) | 1.7252e+0 (5.88e-2)(+) | 1.7982e+0 (7.57e-2)(+) |
| F11-15 | **8.9365e-1(1.51e-2)** | 6.8030e+0 (4.07e+0)(+) | 2.5675e+1 (2.77e-1)(+) | 4.6837e+0 (4.21e-1)(+) | 1.9912e+1 (2.71e+0)(+) |
| F12-15 | **6.4346e+0(1.83e-1)** | 8.0325e+0 (5.71e-2)(+) | 8.5474e+0 (2.94e-1)(+) | 6.5516e+0 (1.13e-1)(+) | 7.0894e+0 (2.10e-1)(+) |
| F13-15 | 1.4841e-1(4.34e-2) | 2.8986e-1 (4.22e-2)(+) | 2.8823e-1 (3.23e-2)(+) | **1.1691e-1 (1.15e-2)(-)** | 1.0959e+0 (3.59e-1)(+) |
| F14-15 | 6.1245e-1(4.40e-2) | 1.1236e+0 (1.29e-1)(+) | **5.1146e-1 (5.54e-3)(-)** | 1.4711e+1 (6.85e+0)(+) | 8.1407e-1 (1.53e-1)(+) |
| F15-15 | 1.3548e+0(4.48e-2) | 3.7230e+0 (0.00e+0)(+) | **1.1226e+0 (1.92e-2)(-)** | 1.1760e+2 (7.36e+0)(-) | 1.1599e+0 (2.94e-2)(-) |
| +/=/- | | 38/2/5 | 40/0/5 | 36/3/6 | 39/0/6 |

注：“+”表示 MaOEA-FM 的性能优于对比算法；“=”表示 MaOEA-FM 的性能与对比算法一样；“-”表示 MaOEA-FM 的性能劣于对比算法